Annals of Operations Research 35(1992)155–186

PERFORMANCE EVALUATION OF POLLING SYSTEMS BY MEANS OF THE POWER-SERIES ALGORITHM

J.P.C. BLANC
Faculty of Economics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

Abstract

Polling systems are widely used to model communication networks with several classes of messages, a single transmission channel and a collision-free access protocol. However, they can only be analysed exactly for some special service disciplines. The power-series algorithm provides a means for the numerical analysis of polling systems with a moderate number of stations, for a wide variety of access protocols. This paper contains a general description of the power-series algorithm, with emphasis on the application to a general class of polling systems with Poisson arrival streams, with Coxian service and switching time distributions, with infinite buffers, with a fixed periodic visit order, and with a Bernoulli schedule for each visit to a station. The applicability and the complexity of the algorithm are discussed for several more service disciplines for polling systems.

1. Introduction

Polling systems are queueing models in which several classes of jobs, tasks or messages compete for service by a single unit, in which access to the service unit is granted according to a collision-free protocol, and in which the times required for switching service from one queue to another may be non-negligible. They are widely used to model computer and communication systems. Examples of applications of polling systems are computer–terminal communication systems, in which jobs are generated by the users of the terminals and in which the central computer examines the terminals and allows them to transmit data one at a time, and local area networks (LANs) consisting of several stations (the queues) connected by a single communication channel (the service unit), in which a token is passed among the stations and in which only the station which is in the possession of the token is allowed to transmit messages (the jobs) over the channel. For surveys on polling systems, the reader is referred to Takagi [25,26], where available solution techniques for these models are reviewed, and to the paper by Levy and Sidi [22], which is more oriented towards modeling and application aspects.

Only few exact results have been obtained for polling systems. The most general type of results consists of pseudo-conservation laws, expressions for a weighted sum of the mean waiting times at the various queues (cf. Boxma [9]). More detailed results have only been obtained for polling systems with specific

service disciplines such as fixed priorities attached to the queues (cf. Jaiswal [18], Klimov and Mishkoy [20]), and periodic polling orders with either exhaustive or gated service at each queue (cf. Cooper [15], Eisenberg [16], Baker and Rubin [1], Sarkar and Zangwill [23]). Therefore, it is useful to develop algorithms for the numerical evaluation of queueing characteristics for a more general class of polling systems. The power-series algorithm (PSA) is a new tool for the performance evaluation of moderately sized multi-queue systems which can be modeled as multi-dimensional quasi-birth–death processes. Such processes consist of a vector of components which describe the number of jobs in each queue, and possibly one or more supplementary components with a finite range, which may be used, for instance, to model Coxian service time distributions. Polling systems form an important class of systems which can be modeled by such a process. The basic idea of the PSA is the transformation of the non-recursively solvable (infinite) set of balance equations for the stationary state probabilities into an, in principle, recursively solvable set of equations by adding one dimension to the state space. This transformation is realized by means of power-series expansions of the state probabilities as functions of the occupancy (load) of a system in light traffic. The basic idea of the PSA has been introduced in Hooghiemstra et al. [17]. The algorithm has been further developed by Blanc [3–6,8], which has led to more efficient implementations of the algorithm, faster convergence of the power series by means of the epsilon algorithm (Wynn [28]), especially when the occupancy of a system is high, and a broader scope of applications for the algorithm. The PSA has been applied to several types of multi-queue models such as the shortest-queue model (Blanc [2,6]), the coupled-processor model (Hooghiemstra et al. [17], Blanc [4]), and some polling models (Blanc [5,7,8]). The PSA provides accurate data for moderate sized systems, which are of interest in themselves for studying the interaction between queues, and which may be helpful in finding and validating approximations for large-scale systems.

The aim of the present paper is to give a survey of various aspects related to the PSA, with emphasis on the application to polling systems. The paper generalizes and unifies previous results on this topic, discussed in Blanc [5,7,8]. First, a general description of the principle and the applicability of the PSA will be given (section 2). Then, the complexity of the computations of the PSA when applied to polling systems with various service disciplines will be discussed (section 3). Next, the PSA will be considered in more detail for polling systems with Poisson arrival streams, with Coxian service and switching time distributions, with a general periodic visit order, and with a Bernoulli schedule for each visit to a station (section 4). Finally, potential further extensions of the models and the algorithm will be indicated in section 5. Polling systems with Bernoulli schedules were previously considered by Servi [24] and Tedijanto [27]. This class of disciplines includes exhaustive and 1-limited service, and may be used as an approximation to K-limited service ($K > 1$). Properties of systems with limited service and Bernoulli schedules (with zero switching times) have been compared in Blanc [7]. The Markovian nature of Bernoulli schedules causes systems with such disciplines to be easier to

analyse than systems with limited service, in particular when applying the PSA (see section 3.2). Another advantage of the class of Bernoulli schedules over the class of limited service disciplines is that the first one is richer because it uses real-valued parameters. A disadvantage of Bernoulli schedules may be their random nature, which will lead to larger standard deviations of performance measures than with comparable limited service disciplines.

2. The power-series algorithm

The PSA will be discussed in the following sections for a general class of models. Topics include conditions for application of the algorithm, means for accelerating the convergence of the series, and notes on implementation of the algorithm.

2.1. DESCRIPTION OF THE MARKOV PROCESS

Consider the following class of multi-dimensional quasi-birth–death processes. The process consists of an s-dimensional vector $N := (N_1, \ldots, N_s)$ and a supplementary variable F. The components of the vector N, which describe the number of jobs in the various queues in steady state, take nonnegative integer values, and the variable F may take a finite number of values, say in the set Θ. In fact, the supplementary space does not need to be the same for each $n \in \mathbb{N}^s$. However, the maximum of the sizes of these spaces over all n, $n \in \mathbb{N}^s$, should be finite. For simplicity of the discussion, we will assume that the supplementary space is the same for each n, while it is possible that some states (n, φ) cannot be entered. The size of the set Θ will be denoted by $|\Theta|$. The stochastic process (N, F) is a stationary Markov process of which each component N_j, $j = 1, \ldots, s$, has a birth–death structure. This means that the time until a transition occurs from a state (n, φ) to some other state is negative exponentially distributed, and that one-step transitions are only possible to states with at most one unit more or one unit less in one of the first s entries. The one-step transition rates are defined to be: for $(n, \varphi) \in \mathbb{N}^s \times \Theta$, $j = 1, \ldots, s$, $\psi \in \Theta$,

$\chi\, a_j(n, \varphi, \psi)$: arrival rate to queue j at state (n, φ), leading to a transition to the state $(n + e_j, \psi)$;

$d_j(n, \varphi, \psi)$: the departure rate from queue j at state (n, φ), leading to a transition to the state $(n - e_j, \psi)$, with $d_j(n, \varphi, \psi) = 0$ if $n_j = 0$;

$u(n, \varphi, \psi)$: the phase-transition rate from state (n, φ) to state (n, ψ).

Here, $e_j \in \mathbb{N}^s$ is the vector with zero entries except an entry of one at the jth position ($j = 1, \ldots, s$), and χ is a parameter which will be used as a variable in power-series expansions; the relative arrival rates $a_j(n, \varphi, \psi)$ are assumed to be normalized such that the systems are stable for $\chi < 1$.

2.2. BALANCE EQUATIONS

The state probabilities are defined as follows: for $(n, \varphi) \in \mathbb{N}^s \times \Theta$,

$$p(n, \varphi) := \Pr\{(N, F) = (n, \varphi)\}. \tag{2.1}$$

The description of the process in the previous section implies that the state probabilities (2.1) satisfy the following balance equations, for $(n, \varphi) \in \mathbb{N}^s \times \Theta$,

$$\left\{ \sum_{j=1}^{s} \sum_{\psi \in \Theta} [\chi a_j(n, \varphi, \psi) + d_j(n, \varphi, \psi)] + \sum_{\psi \in \Theta} u(n, \varphi, \psi) \right\} p(n, \varphi)$$

$$= \chi \sum_{j=1}^{s} \sum_{\psi \in \Theta} a_j(n - e_j, \psi, \varphi) p(n - e_j, \psi) I\{n_j > 0\}$$

$$+ \sum_{j=1}^{s} \sum_{\psi \in \Theta} d_j(n + e_j, \psi, \varphi) p(n + e_j, \psi) + \sum_{\psi \in \Theta} u(n, \psi, \varphi) p(n, \psi). \tag{2.2}$$

Here, $I\{E\}$ stands for the indicator function of the event E. The infinite set of linear equations (2.2) can not, in general, be solved recursively because the equation for $p(n, \varphi)$, $(n, \varphi) \in \mathbb{N}^s \times \Theta$, involves terms with $p(n + e_j, \psi)$, $j = 1, \ldots, s$, $\psi \in \Theta$, and because there exist no local balance relations for most models in the class of models that will be considered.

2.3. CONDITIONS FOR APPLICATION OF THE ALGORITHM

The solution method for the set of equations (2.2) on which the standard PSA is based relies on the following property of the state probabilities:

$$p(n, \varphi) = O(\chi^{n_1 + \cdots + n_s}), \quad \text{as} \quad \chi \downarrow 0, \quad \text{for } (n, \varphi) \in \mathbb{N}^s \times \Theta. \tag{2.3}$$

This property can be shown to hold on the following conditions: for each state $(n, \varphi) \in \mathbb{N}^s \times \Theta$, $n \neq 0$, either $p(n, \varphi) = 0$ (the state cannot be entered), or there exists a path $\vartheta_0, \vartheta_1, \ldots, \vartheta_v$ in Θ for some v, $0 \leq v < |\Theta|$ such that $\vartheta_0 = \varphi$, $u(n, \vartheta_{i-1}, \vartheta_i) > 0$ for $i = 1, \ldots, v$, and

$$\sum_{j=1}^{s} \sum_{\psi \in \Theta} d_j(n, \vartheta_v, \psi) > 0; \tag{2.4}$$

i.e. for each reachable n, $n \neq 0$, there must be at least one positive departure rate, and for each reachable state $(n, \varphi) \in \mathbb{N}^s \times \Theta$, $n \neq 0$, the probability that a departure occurs before any arrival takes place, after the process has entered this state, must

be positive. The proof of this assertion relies on induction to the sum of the queue lengths $n_1, \ldots, n_s$ and to the length of the path v. It is an extension of the proof of a stronger condition than the one above, which has been discussed in Blanc [3], and which requires that for each reachable state $(n, \varphi) \in \mathbb{N}^s \times \Theta$, $n \neq 0$, there is at least one positive departure rate (i.e. $v = 0$ in (2.4) for each φ). However, the latter condition does not hold in polling systems for states in which the server is switching. The above condition (2.4) is fulfilled for polling systems which are usually considered for modeling computer-communication systems, but it is not, for instance, if service only starts when the number of jobs in a queue has reached some threshold larger than 1. It will be clear that (2.4) cannot hold for empty states $(0, \varphi)$, $\varphi \in \Theta$. For these specific states, property (2.3) trivially holds. However, these states require a special treatment, as will be seen in the next section.

2.4. THE COMPUTATION SCHEME

The computation scheme of the PSA for the class of models described in section 2.1 will be given here in its simplest form. A more complicated form may be needed in order to avoid numerical instabilities. These matters will be discussed in the next section. Based on property (2.3), the following power-series expansions are introduced: for $(n, \varphi) \in \mathbb{N}^s \times \Theta$,

$$p(n, \varphi) = \chi^{n_1 + \cdots + n_s} \sum_{k=0}^{\infty} \chi^k b(k; n, \varphi). \tag{2.5}$$

When the power-series (2.5) have been substituted into the balance equations (2.2), then equating the coefficients of corresponding powers of χ in the resulting equations leads to the following set of equations for computing the coefficients of the power-series (2.5): for $(n, \varphi) \in \mathbb{N}^s \times \Theta$, for $k = 0, 1, 2, \ldots$,

$$\left\{ \sum_{\psi \in \Theta} \left[u(n, \varphi, \psi) + \sum_{j=1}^{s} d_j(n, \varphi, \psi) \right] \right\} b(k; n, \varphi) = \sum_{\psi \in \Theta} u(n, \psi, \varphi) b(k; n, \psi)$$

$$+ \sum_{j=1}^{s} \sum_{\psi \in \Theta} \left[a_j(n - e_j, \psi, \varphi) b(k; n - e_j, \psi) I\{n_j > 0\} \right.$$

$$\left. - a_j(n, \varphi, \psi) b(k-1; n, \varphi) I\{k > 0\} \right]$$

$$+ \sum_{j=1}^{s} \sum_{\psi \in \Theta} d_j(n + e_j, \psi, \varphi) b(k-1; n + e_j, \psi) I\{k > 0\}. \tag{2.6}$$

This set of eqs. (2.6) forms a recursive scheme with respect to the components $(k; n)$. In order to make this observation more precise, we introduce the following partial ordering $\prec$ of vectors $(i; m, \psi)$, $(k; n, \varphi) \in \mathbb{N}^{1+s} \times \Theta$:

$$(i;m,\psi) \prec (k;n,\varphi) \quad \text{if } i+m_1+\ldots+m_s < k+n_1+\ldots+n_s;$$

$$\text{or if } i+m_1+\ldots+m_s = k+n_1+\ldots+n_s \text{ and } i<k. \quad (2.7)$$

It is readily verified that the set of eqs. (2.6) expresses coefficients $b(k;n,\varphi)$ in terms of coefficients with a lower order with respect to $\prec$ with the exception of the terms with $b(k;n,\psi)$, $\psi \in \Theta$. Hence, the set of eqs. (2.6) can be divided into subsets of at most $|\Theta|$ equations with unknowns $b(k;n,\varphi)$, $\varphi \in \Theta$. The same conditions which guarantee that (2.3) holds, cf. (2.4), also guarantee that these subsets of equations possess a unique solution. The only exceptions are formed by the empty states, i.e. states with $n=0$, for which all departure rates vanish so that eqs. (2.6) reduce to: for $\varphi \in \Theta$, $k=0,1,2,\ldots$,

$$\sum_{\psi\in\Theta} u(0,\varphi,\psi)b(k;0,\varphi) = -\sum_{j=1}^{s}\sum_{\psi\in\Theta} a_j(0,\varphi,\psi)b(k-1;0,\varphi)I\{k>0\}$$

$$+\sum_{j=1}^{s}\sum_{\psi\in\Theta} d_j(e_j,\psi,\varphi)b(k-1;e_j,\psi)I\{k>0\}+\sum_{\psi\in\Theta} u(0,\psi,\varphi)b(k;0,\psi). \quad (2.8)$$

It is readily verified by summing eqs. (2.8) over φ, $\varphi \in \Theta$, that these are dependent sets of equations for the coefficients $b(k;0,\varphi)$, $\varphi \in \Theta$, for each k, $k=0,1,2,\ldots$. To this end, it should be noted that for each k, $k=0,1,2,\ldots$,

$$\sum_{\varphi\in\Theta} b(k;0,\varphi)\sum_{j=1}^{s}\sum_{\psi\in\Theta} a_j(0,\varphi,\psi) = \sum_{j=1}^{s}\sum_{\psi\in\Theta} b(k;e_j,\psi)\sum_{\varphi\in\Theta} d_j(e_j,\psi,\varphi),$$

because of a necessary balance in transitions between the set of empty states and the set of states with one job in the system. In order to obtain an additional equation, the law of total probability can be used. Substituting (2.5) into the law of total probability gives:

$$\sum_{\varphi\in\Theta} b(0;0,\varphi) = 1,$$

$$\sum_{\varphi\in\Theta} b(k;0,\varphi) = -\sum_{0<n_1+\ldots+n_s\le k}\ \sum_{\psi\in\Theta} b(k-n_1-\ldots-n_s;n,\psi), \quad k=1,2,\ldots. \quad (2.9)$$

Note that the right-hand sides of (2.9) contain only coefficients of states of a lower order with respect to $\prec$ than $(k;0,\varphi)$. Now, all but one of the equations (2.8) together with (2.9) determine $b(k;0,\varphi)$, $\varphi \in \Theta$, for $k=0,1,2,\ldots$, if the Markov process (N,F), conditioned to the event that $N=0$ and no arrivals occur, is an irreducible Markov process on a subset of Θ (note that the determinants of these sets of equations do not depend on k, $k=0,1,\ldots$, so that it is sufficient to consider the solvability of the set of equations for $k=0$).

The PSA is most efficient when the coefficients $b(k; n, \varphi)$ can be computed recursively from (2.6) and (2.9). Whether this is possible depends on the structure of the supplementary space. The computation scheme is recursive if for each $n \in \mathbb{N}^s$ the values of the supplementary variable F can be ordered such that transitions without N leaving the state n are only possible in one direction. More formally, if $\prec_n$ is an ordering of the set Θ, then it should hold that

$$u(n, \varphi, \psi) = 0, \quad \text{if} \quad \psi \prec_n \varphi, \quad \text{for all} \quad \varphi, \psi \in \Theta. \tag{2.10}$$

The ordering does not need to be the same for all $n \in \mathbb{N}^s$. If no ordering with the property (2.10) exists for some n, then a set of at most $|\Theta|$ linear equations has to be solved for such an n. Therefore, Coxian distributions are much easier to handle with the PSA than more general phase-type distributions. Once the coefficients of the power-series expansions of the state probabilities have been determined, those of the moments of the joint queue length distribution can be obtained as well. Write

$$E\left\{\prod_{j=1}^{s} N_j^{v_j}\right\} = \sum_{k=0}^{\infty} \chi^k f(k; v) \qquad v \in \mathbb{N}^s. \tag{2.11}$$

It follows from (2.11) and (2.5) that for $v \in \mathbb{N}^s$, $k = 0, 1, 2, \ldots$,

$$f(k; v) = \sum_{0 \le n_1 + \ldots + n_s \le k} \sum_{\varphi \in \Theta} \prod_{j=1}^{s} n_j^{v_j} b(k - n_1 - \ldots - n_s; n, \varphi). \tag{2.12}$$

It is more convenient for obtaining moments of the queue length distribution to compute first their coefficients via (2.12) and then to use (2.11) than to compute first the state probabilities via (2.5) and then the moments directly from the state probabilities. In the first way, algorithms for accelerating the convergence can be applied to partial sums of the series (2.11), and the storage requirement for the coefficients can be reduced; see sections 2.5 and 2.6.

2.5. CONVERGENCE OF THE POWER-SERIES

Experience has taught us that the power-series (2.5) and (2.11) usually do not converge for all values of χ for which a system is stable (by definition, for $\chi < 1$). One way to overcome this difficulty is to introduce the following bilinear mapping of the interval [0, 1] onto itself,

$$\vartheta = \frac{(1+G)\chi}{1+G\chi}, \quad G \ge 0. \tag{2.13}$$

This transformation maps possible singularities of the power-series in the region

$$\left|\chi - \frac{G}{1+2G}\right| > \frac{1+G}{1+2G}, \quad |\chi| \le 1, \tag{2.14}$$

outside the unit circle in the complex ϑ-plane. Another computation scheme is then obtained by introducing power-series expansions of the state probabilities as functions of ϑ, by replacing χ by ϑ in the balance equations (2.2) according to (2.13), and by substituting the power-series in ϑ into these equations. Equating coefficients of corresponding powers of ϑ in the resulting equations leads to a set of equations which differ from (2.6) mainly through the occurrence of terms with coefficients of the form $b(k-2; n+e_j, \psi)$. Any singularity outside the circle $|\chi - 1/2| = 1/2$ may be removed from the unit disk by this procedure with an appropriate choice of the parameter G.

Another technique for removing singularities from inside the unit disk is application of the epsilon algorithm or a related algorithm. The epsilon algorithm aims to accelerate the convergence of slowly convergent sequences or to determine a value for divergent sequences (cf. Wynn [28], Brezinski [12]). It converts a polynomial into a quotient of two polynomials. The epsilon algorithm consists of the following triangular recursive scheme: for $m = 0, 1, \ldots, \kappa = 0, 1, \ldots,$

$$\varepsilon_{\kappa+1}^{(m)} = \varepsilon_{\kappa-1}^{(m+1)} + \left[\varepsilon_{\kappa}^{(m+1)} - \varepsilon_{\kappa}^{(m)}\right]^{-1}, \quad \varepsilon_{-1}^{(m)} = 0, \ \varepsilon_{0}^{(m)} = S_m; \tag{2.15}$$

here, the initial sequence S_m, $m = 0, 1, \ldots,$ consists of partial sums of a series. Only the even sequences $\{\varepsilon_{2\kappa}^{(m)}, m = 0, 1, \ldots\}$, $\kappa = 1, 2, \ldots,$ may be sequences which converge faster to a limit than the initial sequence. The odd sequences are only intermediate steps in the calculation scheme. When S_m is the partial sum of a power-series, say

$$S_m = S_m(\chi) = \sum_{k=0}^{m} c_k \chi^k, \quad m = 0, 1, \ldots, \tag{2.16}$$

then the epsilon algorithm transforms this sequence of polynomials into sequences of quotients of two polynomials. More precisely, $\varepsilon_{2\kappa}^{(m-2\kappa)}$ will be a quotient of a polynomial of degree $m - \kappa$ over a polynomial of degree κ, and

$$\left|S_m - \varepsilon_{2\kappa}^{(m-2\kappa)}\right| = O(\chi^{m+1}), \quad \chi \to 0, \ \kappa = 1, 2, \ldots, \ m = 2\kappa, 2\kappa+1, \ldots; \tag{2.17}$$

(see Wynn [28]). When the heavy traffic behaviour of the moments of the queue length distribution is known beforehand, the performance of the epsilon algorithm can be improved by a modification of the initial values $\varepsilon_0^{(m)}$ (cf. Blanc [5]). Before application of the epsilon algorithm, the coefficients of the power-series are extrapolated to take into account the pole at $\chi = 1$. This means that we take for first-order poles

$$\varepsilon_0^{(m)} = S_m + c_m \frac{\chi^{m+1}}{1-\chi}, \qquad m = 1, 2, \ldots, \tag{2.18}$$

and for second-order poles

$$\varepsilon_0^{(m)} = S_m + \left[c_m + \frac{c_m - c_{m-1}}{1-\chi} \right] \frac{\chi^{m+1}}{1-\chi}, \quad m = 1, 2 \ldots, \tag{2.19}$$

instead of the last relation of (2.15); here, S_m is of the form (2.16) and the c_k, $k = 0, 1, \ldots$, stand for coefficients of a series as defined in (2.11). The pole at $\chi = 1$ is preserved in other (even) sequences produced by the epsilon algorithm. It should be noted that not every queue grows without bounds as $\chi \uparrow 1$ in some systems (see, e.g. the discussion in section 4.5); modifications (2.18) and (2.19) should only be applied to those moments which do have a pole at $\chi = 1$ in order to accelerate the convergence, although $\varepsilon_0^{(m)}$ defined by (2.18), respectively (2.19), will converge to the same limit as S_m if the latter sequence is convergent.

The epsilon algorithm turns a divergent series into a convergent series if the analytic continuation of the function defined by the series at $\chi = 0$ possesses only a finite number of poles as singularities inside the unit circle $|\chi| \leq 1$. The latter seems to hold for all models considered. However, it may happen that the power series are so strongly divergent that numerical instabilities occur when a large number of terms is computed. In that case, a conformal mapping as discussed above, cf. (2.13), should be used together with the epsilon algorithm. The value of G in the mapping (2.13), the number of terms M of the power-series expansions, and the number of steps κ in the epsilon algorithm, cf. (2.15), which are needed to reach a certain accuracy, depend on various properties of the models. Generally, these quantities increase with increasing load, with increasing number of queues, and with increasing asymmetry between the parameters of the various queues. Numerical experience has taught us that application of the epsilon algorithm strongly improves the performance of the PSA and that, in many cases, it even leads to good estimations of heavy traffic limits. To illustrate the above-discussed properties of the transformation (2.13) and of the epsilon algorithm, values of $\varepsilon_{2\kappa}^{(m-2\kappa)}$ have been listed in table 1 for various values of G, m and κ, for a specific quantity (namely, $E\{N_1 + \ldots + N_s\}$ for the case Dd1 of section 4.7). For each value of m and G the value of κ, indicated by κ_{opt}, has been determined for which the following difference was minimal:

$$\max_{i=0,\ldots,\alpha} \left\{ \varepsilon_{2\kappa}^{(m-2\kappa-i)} \right\} - \min_{i=0,\ldots,\alpha} \left\{ \varepsilon_{2\kappa}^{(m-2\kappa-i)} \right\};$$

here, we have chosen $\alpha = 1$ for $m = 6$ and $m = 12$, $\alpha = 3$ for $m = 20$ and $\alpha = 5$ for $m = 36$. Note that increasing α implies that a smaller number of iterations of the epsilon algorithm can be executed. The values corresponding to κ_{opt} have been

Table 1

Performance of the epsilon algorithm

m	G	κ: 0	1	2	m	G	κ: 0	1	2	3	4	5
6	0.0	10.97	9.752	8.851	12	0.0	−846	11.04	9.396	9.807	9.804	9.797
6	0.5	9.820	9.820	7.347	12	0.5	7.571	9.746	9.829	9.815	9.798	9.795
6	0.5	9.880	9.247	−5.087	12	1.0	9.783	9.814	9.875	9.803	9.794	9.798

m	G	κ: 0	1	2	3	4	5	6	7	8
20	0.0	-4×10^{6}	1×10^{5}	−1473	10.72	9.7827	9.8117	9.79270	9.79262	9.79182
20	0.5	229	−19.8	7.988	9.796	9.7928	9.7961	9.79593	9.79196	9.79186
20	1.0	9.465	9.885	9.794	9.796	9.7932	9.7917	9.79219	9.79218	9.79104

m	G	κ: 0	1	2	3	4	5	6	opt	(κ_{opt})	15
36	0.0	-2×10^{15}	1×10^{14}	5×10^{11}	-2×10^{9}	2×10^{6}	3×10^{6}	−569	9.79193	(11)	9.79193
36	0.5	2×10^{8}	-2×10^{7}	-2×10^{4}	-3×10^{4}	49.1	8.624	9.76330	9.79192	(14)	9.79191
36	1.0	−541	175	9.889	9.898	9.791	9.792	9.79198	9.79191	(13)	9.79191
36	1.5	9.771	9.771	9.792	9.792	9.792	9.792	9.79192	9.79191	(14)	9.79191
36	2.0	9.792	9.792	9.792	9.792	9.792	9.792	9.79211	9.79191	(13)	9.79198
36	2.5	9.793	9.792	9.792	9.792	9.792	9.792	9.79191	9.79192	(7)	9.79193

Table 2

Accuracy of the algorithm; CPU times

m	G	PCL	CPU	G	PCL	CPU	G	PCL	CPU
6	0.0	3.72372	0:01	0.5	3.55850	0:01	1.0	3.62983	0:01
12	0.0	3.79617	0:09	0.5	3.79787	0:10	1.0	3.82088	0:10
20	0.0	3.79675	1:24	0.5	3.79518	1:25	1.0	3.79676	1:28
36	0.0	3.79741	20:26	0.5	3.79741	20:57	1.0	3.79739	20:53
36	1.5	3.79740	21:21	2.0	3.79739	20:58	2.5	3.79735	21:34

underlined in table 1. The above criterion usually provides good estimates for the performance measures. As a further check on the accuracy of the algorithm, table 2 contains the values of the left-hand sides of the pseudo-conservation law PCL (to be discussed in section 4.5, cf. (4.30)) for the same model as in table 1, based on the values of $E\{N_j\}$ with $\kappa = \kappa_{opt}$ for each j, $j = 1, \ldots, s$; for comparison, the known right-hand side of the pseudo-conservation law is 3.79746 for this example. Table 2 also contains the CPU times (displayed as minutes : seconds) which were required for the computations on a VAX-8700. Note that the required CPU times are subject to some randomness, because the amount of arithmetical operations is exactly the same for fixed m and $G > 0$.

2.6. ON THE IMPLEMENTATION OF THE ALGORITHM

For most models, limitations on storage capacity are more important restrictions on the applicability of the PSA than limitations on computing time. The complexity of the PSA mainly depends on the number of stations s and on the size of the supplementary space Θ. The evaluation of power-series expansions up to the Mth power of χ requires the computation of

$$\binom{M+s+1}{s+1}|\Theta| \tag{2.20}$$

coefficients $b(k; n, \varphi)$, namely those with $k + n_1 + \ldots + n_s \leq M$, $\varphi \in \Theta$. The complexity of the computation of a single coefficient $b(k; n, \varphi)$ depends on the structure of the model, in particular on the number of non-zero transition rates. If all transition rates which occur in eq. (2.6) are positive and if all entries $k, n_1, \ldots, n_s$ are positive, then the computation of one coefficient $b(k; n, \varphi)$ requires $1 + (2s + 1)|\Theta|$ multiplications, $(4s + 2)|\Theta|$ additions and subtractions, and 1 division. When the transformation (2.13) is used, then some additional terms appear in the recurrence relations, cf. Blanc [5], and the number of operations increases to $3 + (3s + 2)|\Theta|$ multiplications, $(5s + 3)|\Theta|$ additions and subtractions, and 2 divisions. However, for most (polling) models, many transition rates vanish so that the number of operations will be less; see section 4.4 for an example.

In order to make an efficient use of the available memory space, we map the multi-dimensional region of lattice points $k + n_1 + \ldots + n_s \leq M$ onto the set of integers by means of the one-to-one mapping (cf. Blanc [5]),

$$C(k;n) = \sum_{j=0}^{s} \binom{k+j+\sum_{i=1}^{j} n_i}{j+1}. \tag{2.21}$$

This procedure enlarges the number of terms of the power-series expansions which can be computed with a given storage capacity at the cost of increased computation time needed for the determination of the location of the coefficients in the array in which they are stored. A further reduction of storage requirement can be achieved when only a limited number of performance measures have to be evaluated. In most cases, one is not interested in all individual state probabilities. Then, the coefficients of the power-series expansions of the important performance measures can be aggregated during the execution of the PSA, cf., for example, (2.12), and stored in separate (relatively small) arrays, while the coefficients of the state probabilities can be deleted as soon as they are not needed anymore in further computations. This approach reduces the storage requirement for calculating M terms of the power-series expansions to $D_M \times |\Theta|$, where D_M is the largest distance (in terms of the mapping $C(k; n)$, cf. (2.21)) between coefficients occurring in a single equation of (2.6) (cf. Blanc [5]),

$$D_M = \binom{M+s}{s}, \text{ if } G = 0; \quad D_M = \binom{M+s}{s} + \binom{M+s-2}{s-1}, \text{ if } G > 0. \tag{2.22}$$

Table 3 shows as an illustration the maximal number of terms M which can be obtained with a storage capacity of 2×10^6 coefficients according to (2.22) with $G = 0$, as a function of the number of stations s and the size of the supplementary space Θ. The maximal value of M in the case $G > 0$ is at most one less than in the

Table 3

The maximal number of terms M at a storage capacity of 2×10^6

s	$\lvert\Theta\rvert$: 4	8	12	16	20	24	28	32	36	40	44	48
2	998	705	575	498	445	406	376	352	331	314	300	287
4	56	47	42	39	36	35	33	32	31	30	29	29
6	23	20	18	17	16	16	15	15	15	14	14	14
8	15	13	12	11	11	11	10	10	10	10	10	9
10	11	10	9	9	9	8	8	8	8	8	8	7
12	9	8	8	7	7	7	7	7	7	6	6	6

case $G = 0$. While the above procedures reduce storage requirement and increase programming flexibility with respect to the number of stations, they add to the computational burden. When using the procedure to compute the values of the

mapping $C(k; n)$ in (2.21) simultaneously for a state and its neighbouring states described in Blanc [5], $s(s-1)$ multiplications and divisions are required for each vector $(k; n)$ if $G = 0$ and $s(s+1)$ if $G > 0$. Combining the results of this section, we obtain the following upper bounds on the number of operations (multiplications and divisions only) required for computing the coefficients of the power-series expansions of the state probabilities up to the Mth power of χ:

$$\binom{M+s+1}{s+1}\left[s(s-1)+|\Theta|\{2+(2s+1)|\Theta|\}\right], \quad \text{if } G = 0;$$

$$\binom{M+s+1}{s+1}\left[s(s+1)+|\Theta|\{5+(3s+2)|\Theta|\}\right], \quad \text{if } G > 0. \tag{2.23}$$

Finally, we note that the time required to compute performance measures, given the coefficients $b(k; n, \varphi)$, is negligible compared to the time required to compute these coefficients themselves.

3. Polling systems

Models for polling systems usually consist of several stations (queues), each with an arrival stream of jobs, which are attended to by a single server. The server visits the stations according to some control rule (service discipline). In many cases, switch-over times or set-up times are required when the server changes service from one queue to another. Important areas for application of these models are computer-communication systems, in which several stations share a common communication channel and compete for access to this channel, and manufacturing systems, in which several types of products have to be manufactured on a single production unit.

3.1. SERVICE DISCIPLINES

The service disciplines for polling systems can often be divided into three parts, which can be chosen independently of each other:

A. a rule for the order in which the server visits the queues;

B. rules for the number of services per visit to the various queues;

C. a rule for the behaviour of the server when the system is empty.

Examples of order-of-visit rules are:

A1. polling in a fixed periodic order (cyclic: $1, 2, \ldots, s$, star: $1, 2, 1, 3, \ldots, 1, s$, scanning: $1, 2, \ldots, s, s, s-1, \ldots, 1$, or according to some general polling table);

A2. random or Markovian polling: the next queue to be visited is determined by a random mechanism which may depend on the current position of the server (Markovian polling, cf. Boxma and Weststrate [11]) or not (random polling, cf. Kleinrock and Levy [19]);

A3. polling according to fixed priorities attributed to the queues: the next queue to be visited is the non-empty queue with the highest priority, cf. Jaiswal [18], Klimov and Mishkoy [20];

A4. polling according to a dynamic (state-dependent) rule such as priority for the longest queue, cf., for example, Cohen [14].

The choice of the order-of-visit rule will depend on the availability of information about the presence of jobs at the various stations. Rules A1 and A2 require only local information, about the station where the server is present, but rules A3 and A4 require also information from other stations. Further, this choice may depend on the configuration of the system, i.e. on the existence or non-existence of a direct connection between pairs of stations in the network, and on the distances between the stations, in terms of mean switching times.

Examples of number-of-services rules are:

B1. exhaustive service (the server remains serving until a queue becomes empty);

B2. gated service (all jobs present in a queue at the instant at which the server arrived at that queue are served);

B3. limited service (a fixed number of jobs are served, at most);

B4. Bernoulli service (after each service, another service may be started with a fixed probability, cf. Servi [24], Tedijanto [27]);

B5. semi-exhaustive service (the server attends to a queue until the number of jobs in that queue has become one less than the number of jobs in that queue at the instant at which the server arrived at that queue, cf. Cohen [13]).

The number-of-services rules may be different for the various queues, and may even be different at various visits to the same queue (e.g. when a queue occurs more than once on a polling table).

Examples of empty-system rules are:

C1. the server keeps on switching according to the order-of-visit rule;

C2. the server remains at the last served queue;

C3. the server goes to a state of rest;

C4. the server goes to a specific queue (e.g. the queue with the highest load, or the queue with the largest arrival rate).

The choice of the empty-system rule will also depend on the availability of information. Rule C1 requires only local information; in fact, the server does not even have to know that the system is empty. The other rules require information from all stations.

As far as the choices of these rules are not limited by the physical configuration of a system or by the available information, they can be used to optimize the performance of a system with respect to some criteria.

3.2. COMPLEXITY OF THE ALGORITHM

The complexity of the PSA, in the sense of the number of coefficients $b(k; n, \varphi)$ which have to be computed in order to obtain some fixed number M of terms of the power-series expansions, will be discussed in this section for polling systems with various service disciplines. This complexity is given by (2.20), where the size of the supplementary space Θ depends on the service discipline and on the number of phases of the interarrival, service and switching time distributions. The supplementary space has to contain information on the position of the server (in the network, on the polling table, etc.), on the state of the server (switching, serving, etc.), and on the actual phases of the Coxian distributions. Here, the discussion will be confined to Poisson arrival streams at each station. Then, at each instant there is only one distribution of which the actual phase is relevant, because there is only one server in the system. Let $\Psi_j^1 \geq 1$ be the number of phases of the distribution of the service times at queue j, $j = 1, \ldots, s$, and $\Psi_{i,j}^0 \geq 1$ the number of phases of the distribution of the switching times from queue i to queue j, $i, j = 1, \ldots, s$. First, consider polling in a fixed (static) periodic order. Such order-of-visit rules can generally be described by a polling table of some finite length L, $L \geq s$. A polling table can be constructed by a mapping $l: \{1, \ldots, L\} \to \{1, \ldots, s\}$. If the number-of-services rule is limited service for each visit, with limit K_h for the hth entry on the table, $h = 1, \ldots, L$, then the supplementary space consists for each entry h of the phases of the switch from queue $l(h-1)$ to $l(h)$ – here, $l(0) = l(L)$ – and of K_h times the phases of the services at queue $l(h)$, so that its size becomes

$$|\Theta| = \sum_{h=1}^{L} \left[\Psi_{l(h-1),l(h)}^0 + K_h \Psi_{l(h)}^1 \right] \geq 2L \geq 2s. \tag{3.1}$$

The lower bound $2L$ is realized when all service times and switching times are exponentially distributed and all service limits are equal to 1, and the lower bound $2s$ is realized when moreover the polling order is cyclic. Next, consider order-of-visit rules which require only information on the current position of the server and on the number of jobs at the stations to determine the next station to be visited. This set of rules includes Markovian polling, polling with fixed priorities, and polling with priority for the longest queue. If the number-of-services rule is limited service for each visit, with limit K_j for queue j, $j = 1, \ldots, s$, then the size of the supplementary space becomes

$$|\Theta| = \sum_{i=1}^{s} \sum_{j=1}^{s} \Psi_{i,j}^{0} + \sum_{j=1}^{s} K_j \Psi_j^1 \geq s(s-1) + s = s^2. \tag{3.2}$$

The lower bound s^2 is realized when all service times and switching times are exponentially distributed (while we assumed that $\Psi_{i,i}^0 = 0$, $i = 1, \ldots, s$) and all service limits are equal to 1. This lower bound presumes that switches of the server may occur between each pair of stations. The latter does not necessarily hold for Markovian polling; for this discipline, the lower bound is related to the number of non-zero entries in the matrix of transition probabilities for the server. When the distributions of the switching times depend only on the station to which (and not on the station from which) the server is moving, then $\Psi_{i,j}^0 = \Psi_j^0$, $i, j = 1, \ldots, s$, and the size of the supplementary space can be reduced to:

$$|\Theta| = \sum_{j=1}^{s} K_j \Psi_j^1 + \sum_{j=1}^{s} \Psi_j^0 \geq 2s. \tag{3.3}$$

When switching times may be neglected, the size of the required supplementary space can be found by taking $\Psi_{i,j}^0 = 0$, $i, j = 1, \ldots, s$, in (3.1), (3.2), (3.3), where the lower bounds then reduce to s. The contribution to the size of the supplementary space of a station with a Bernoulli schedule (including exhaustive service) is the same as that of a station with 1-limited service. Therefore, systems with Bernoulli schedules are most suitable for application of the PSA. Stations with gated or semi-exhaustive service discipline require, in principle, an unbounded supplementary space, and would therefore not fit in the framework of the class of models described in section 2.1. However, only a finite number of terms (M) of the power-series expansions are computed in practice, which implies that states with more than M jobs in the system are not considered. Consequently, the size of the supplementary space can still be derived from (3.1), (3.2), (3.3) for the various order-of-visit rules when there are stations with gated or semi-exhaustive service, namely by taking M as the service limit for these stations. However, it will be clear that the PSA is not very suitable for the analysis of systems with gated-type service disciplines. It is true that there exist other, more efficient, methods for analyzing systems with gated service (and exhaustive service), cf., for example, Sarkar and Zangwill [23], but they produce mainly mean waiting times, while the PSA also computes state probabilities and higher-order moments. The foregoing argument also implies that it is useless to consider service limits larger than M when computing M terms of the power-series expansions with the PSA, because performance measures for queues with such limits will be the same as when they had been computed with the exhaustive service discipline at those queues. In fact, the queue length process at stations with a limited service discipline is very similar to that at equivalent stations with exhaustive service in light traffic, while the range of the load for which this similarity continues increases with the service limit. Information on the intrinsic heavy traffic characteristics of a queue with service limit K has to be found in coefficients corresponding to

powers of χ higher than K. On the contrary, for a queue with a Bernoulli parameter q, all coefficients corresponding to powers of χ higher than 1 depend on q. As a consequence, systems with Bernoulli schedules are much more easily studied over all parameter values of the service discipline than systems with limited service, at least with the aid of the PSA together with the epsilon algorithm.

The foregoing discussion of the complexity of the PSA refers only to order-of-visit rules and number-of-service rules, and ignores the influence of empty-system rules. In fact, the above observations hold if the behaviour of the server when the system is empty is similar to that when the system is not empty, e.g. for rule C1 and possibly for rule C4, depending on how interruptions of switches when the system is empty and a job arrives are handled. When this behaviour is deviating, then the size of the supplementary space may have to be larger than indicated above. For instance, when rule C3 is applied and if $\Psi_{0,j}^0$ is the number of phases of the distribution of the switching times from the state of rest to queue j, $j = 1, \ldots, s$, while switches to the state of rest are instantaneous, then (3.2) should be modified to

$$|\Theta| = \sum_{j=1}^{s} K_j \Psi_j^1 + \sum_{i=0}^{s} \sum_{j=1}^{s} \Psi_{i,j}^0 \geq s + s^2, \tag{3.4}$$

while (3.3) remains unchanged if, moreover, $\Psi_{0,j}^0 = \Psi_j^0$, $j = 1, \ldots, s$. Finally, note that the process (N, F) restricted to $N = 0$ is not an irreducible Markov process in the case of rule C2 (each reachable empty state forms an irreducible subchain in this case), so that this rule does not fit in the framework of section 2.4.

4. Systems with a polling table and Bernoulli schedules

In this section, the PSA will be discussed in more detail for the class of polling models with infinite buffers, with Poisson arrival streams, with Coxian service and switching time distributions, with a fixed periodic visit order, and with a Bernoulli schedule for each visit. Section 5 contains some remarks on extensions of this class of models.

4.1. DESCRIPTION OF THE MODEL

The system consists of s queues and a single server. Jobs arrive at queue j according to a Poisson process with rate λ_j, $j = 1, \ldots, s$. Each queue may contain an unbounded number of jobs. The server inspects the queues in an order which is determined by a polling table of finite length L. This table will be described by a mapping $l: \{1, \ldots, L\} \to \{1, \ldots, s\}$. Throughout, it will be assumed that each station occurs at least once on the table, and that L has been chosen as small as possible, given a fixed visit order. Further, the convention $l(0) = l(L)$ will be needed and used. Bernoulli schedules will be used to determine the number of services during the visits of the server to the stations. When the server arrives at a queue,

at least one job is served, unless this queue is empty (in which case, the server directly proceeds to the next queue on the polling table). After the completion of a service at queue $l(h)$, the server starts serving another job at this queue with probability q_h if queue $l(h)$ has not yet been emptied; otherwise, the server proceeds to the next queue on the polling table ($h = 1, \ldots, L$). At each station, jobs are served in order of arrival. Service times of jobs arriving at queue j are assumed to be distributed according to a Coxian distribution with νth moment $\gamma_{\nu j}$, $\nu = 1, 2, \ldots$, $j = 1, \ldots, s$. The Coxian service time distribution at station j, $j = 1, \ldots, s$, consists of Ψ_j^1 phases; with probability $\pi_j^{1,\varphi}$, a service is composed of phases $\varphi, \varphi - 1, \ldots, 1$, $\varphi = 1, \ldots, \Psi_j^1$, and the transition rate from phase ψ is $\mu_j^{1,\psi}$, $\psi = 1, \ldots, \Psi_j^1$. Consequently, the Laplace–Stieltjes transform (LST) of the service time distribution at station j, $j = 1, \ldots, s$, is given by

$$\gamma_j(\omega) = \sum_{\varphi=1}^{\Psi_j^1} \pi_j^{1,\varphi} \prod_{\psi=\varphi}^{\Psi_j^1} \frac{\mu_j^{1,\psi}}{\mu_j^{1,\psi} + \omega}, \quad \text{Re } \omega \geq 0. \tag{4.1}$$

The times which are needed for switching from queue $l(h-1)$ to queue $l(h)$ are also assumed to be distributed according to a Coxian distribution, with νth moment $\delta_{\nu h}$, $\nu = 1, 2, \ldots$, and with parameters Ψ_h^0, $\pi_h^{0,\varphi}$, $\mu_h^{0,\varphi}$, $\varphi = 1, \ldots, \Psi_h^0$, which are defined in a similar way as those of the service time distributions, $h = 1, \ldots, L$.

The sum of the arrival processes at the various queues is a Poisson process with rate $\Lambda := \sum_{j=1}^{s} \lambda_j$. The LST of the service time of an arbitrary job is $\gamma_j(\omega)$ with probability λ_j/Λ, $j = 1, \ldots, s$. Hence, the first two moments β_1 and β_2 of the distribution of the service time of an arbitrary job are given by:

$$\beta_1 = \sum_{j=1}^{s} \frac{\lambda_j}{\Lambda} \gamma_{1j}, \quad \beta_2 = \sum_{j=1}^{s} \frac{\lambda_j}{\Lambda} \gamma_{2j}. \tag{4.2}$$

The offered load ρ_j to station j and the offered load ρ to the system are defined by

$$\rho_j := \lambda_j \gamma_{1j}, \quad \rho := \sum_{j=1}^{s} \rho_j = \Lambda \beta_1. \tag{4.3}$$

The first two moments σ_1 and σ_2 of the total switching time during one cycle of the server along the queues according to the polling table are given by:

$$\sigma_1 = \sum_{h=1}^{L} \delta_{1h}, \quad \sigma_2 = \sum_{h=1}^{L} \delta_{2h} + 2 \sum_{h=1}^{L} \sum_{i=1}^{h-1} \delta_{1h} \delta_{1i}. \tag{4.4}$$

4.2. CONDITIONS FOR STABILITY

Kühn [21] has derived general conditions for stability of cyclic polling systems. Similar ideas lead for the present model with Bernoulli schedules and a general periodic polling order to the following conditions:

$$\lambda_j E\{C\} < m_j, \quad \text{for } j = 1, \ldots, s; \tag{4.5}$$

here, $E\{C\}$ stands for the mean cycle time of the server, i.e. the mean time the server needs to go once along the queues in the order listed on the polling table, so that $\lambda_j E\{C\}$ is the mean number of jobs which arrive at queue j during one cycle, and m_j stands for the mean number of jobs at queue j which can be served during one cycle, $j = 1, \ldots, s$. The latter quantities depend on the polling table and on the Bernoulli parameters, and are readily verified to be equal to:

$$m_j = \sum_{h=1}^{L} \frac{I\{l(h) = j\}}{1 - q_h}, \quad j = 1, \ldots, s. \tag{4.6}$$

Here, $m_j := \infty$ whenever there is at least one h, $h = 1, \ldots, L$, with $l(h) = j$ and $q_h = 1$. The mean cycle time can be found by noting that, with V_h, $h = 1, \ldots, L$, the duration of the hth visit in a cycle (to queue $l(h)$):

$$E\{C\} = \sigma_1 + \sum_{h=1}^{L} E\{V_h\}, \tag{4.7}$$

and, by balance arguments for the flow of jobs into and out of the s queues,

$$\lambda_j E\{C\} = \sum_{h=1}^{L} I\{l(h) = j\} E\{V_h\}/\gamma_{1j}, \quad j = 1, \ldots, s. \tag{4.8}$$

Eliminating $E\{V_h\}$, $h = 1, \ldots, L$, from (4.7) with (4.8) readily leads to

$$E\{C\} = \frac{\sigma_1}{1 - \rho}. \tag{4.9}$$

The conditions (4.5) can be summarized in the following condition:

$$\chi := \rho + \sigma_1 \max_{j=1,\ldots,s} \{\lambda_j / m_j\} < 1. \tag{4.10}$$

This condition depends on the service discipline (in this case, the polling table and the Bernoulli schedules). An intrinsic condition for stability of a polling system is $\rho < 1$. If this condition is satisfied, then it is possible to choose a service discipline such that the system is stable, e.g. one with exhaustive service at each station. We will call χ the *occupancy* of the system. Because this quantity χ will be used as variable in power-series expansions, the arrival rates will be written, cf. section 2.1, as

$$\lambda_j = a_j \chi, \quad j = 1, \ldots, s. \tag{4.11}$$

4.3. BALANCE EQUATIONS

For the formulation of the balance equations for the class of polling systems described in section 4.1, it is most appropriate to introduce a triple (H, Z, Φ) of supplementary variables in order to transform the queue length process into a Markov process. The variable H will indicate the actual position on the polling table (the value of H, in the range $1, \ldots, L$, changes at instants at which the server leaves a station), the variable Z will indicate whether the server is switching ($Z = 0$) or serving ($Z = 1$), and the variable Φ will indicate the actual phase of either the switching time or the service time. The state probabilities are defined as follows: for $\boldsymbol{n} \in \mathbb{N}^s$, $h = 1, \ldots, L$, $\zeta = 0, 1$, $\varphi = 1, \ldots, \Psi_h^0$ if $\zeta = 0$, $\varphi = 1, \ldots, \Psi_{l(h)}^1$ if $\zeta = 1$,

$$p(\boldsymbol{n}, h, \zeta, \varphi) := \Pr\{(\boldsymbol{N}, H, Z, \Phi) = (\boldsymbol{n}, h, \zeta, \varphi)\}. \tag{4.12}$$

Noting that the Coxian distributions have been defined such that completion of a service or a switch can only occur if $\Phi = 1$, cf. section 4.1, the balance equations for the state probabilities (4.12) are readily verified to be, for $\boldsymbol{n} \in \mathbb{N}^s$, $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$,

$$\begin{aligned}\left[\chi \sum_{j=1}^{s} a_j + \mu_h^{0,\varphi}\right] p(\boldsymbol{n}, h, 0, \varphi) &= \chi \sum_{j=1}^{s} a_j\, p(\boldsymbol{n} - \boldsymbol{e}_j, h, 0, \varphi) I\{n_j > 0\} \\ &+ \mu_h^{0,\varphi+1} p(\boldsymbol{n}, h, 0, \varphi+1) I\{\varphi < \Psi_h^0\} + \mu_{h-1}^{0,1}\, \pi_h^{0,\varphi} p(\boldsymbol{n}, h-1, 0, 1) I\{n_{l(h-1)} = 0\} \\ &+ \mu_{l(h-1)}^{1,1}\, \pi_h^{0,\varphi} p(\boldsymbol{n} + \boldsymbol{e}_{l(h-1)}, h-1, 1, 1)[1 - q_{h-1} I\{n_{l(h-1)} > 0\}];\end{aligned} \tag{4.13}$$

and for $\boldsymbol{n} \in \mathbb{N}^s$, $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_{l(h)}^1$, $n_{l(h)} > 0$,

$$\begin{aligned}\left[\chi \sum_{j=1}^{s} a_j + \mu_{l(h)}^{1,\varphi}\right] p(\boldsymbol{n}, h, 1, \varphi) &= \chi \sum_{j=1}^{s} a_j\, p(\boldsymbol{n} - \boldsymbol{e}_j, h, 1, \varphi) I\{n_j > 0\} \\ &+ \mu_{l(h)}^{1,\varphi+1} p(\boldsymbol{n}, h, 1, \varphi+1) I\{\varphi < \Psi_{l(h)}^1\} + \mu_h^{0,1}\, \pi_{l(h)}^{1,\varphi}\, p(\boldsymbol{n}, h, 0, 1) \\ &+ q_h \mu_{l(h)}^{1,1}\, \pi_{l(h)}^{1,\varphi}\, p(\boldsymbol{n} + \boldsymbol{e}_{l(h)}, h, 1, 1).\end{aligned} \tag{4.14}$$

It should be noted that for all φ, $\varphi = 1, \ldots, \Psi_{l(h)}^1$,

$$p(\boldsymbol{n}, h, 1, \varphi) = 0, \quad \text{if } n_{l(h)} = 0, \ h = 1, \ldots, L. \tag{4.15}$$

4.4. THE COMPUTATION SCHEME

First, it will be shown that the Markov process (N, H, Z, Φ) satisfies conditions (2.4) and, hence, that the state probabilities (4.12) possess the light traffic behaviour as indicated in (2.3). For this purpose, we introduce for each $n \in \mathbb{N}^s$, $n \neq 0$, an ordering $\prec_n$ of the vectors of supplementary values (h, ζ, φ). For each $n \in \mathbb{N}^s$, $n \neq 0$, we define for each h, $h = 1, \ldots, L$:

$$(h, \zeta_1, \varphi_1) \prec_n (h, \zeta_2, \varphi_2) \quad \text{if} \quad \zeta_1 < \zeta_2, \quad \text{or if} \quad \zeta_1 = \zeta_2 \text{ and } \varphi_1 > \varphi_2, \tag{4.16}$$

i.e. the vectors are ranked in increasing order as

$$(h, 0, \Psi_h^0), (h, 0, \Psi_h^0 - 1), \ldots, (h, 0, 1), (h, 1, \Psi_{l(h)}^1), \ldots, (h, 1, 1). \tag{4.17}$$

For each $n \in \mathbb{N}^s$, $n \neq 0$, there is an i with $n_i > 0$, and by definition of the polling table there exists an $h_n \in \{1, \ldots, L\}$ such that $l(h_n) = i$. The subsets (4.17) of vectors (h, ζ, φ) are ranked with respect to the component h by increasing order of $h_n + 1, \ldots, L, 1, \ldots, h_n$, i.e. for all $\zeta_1, \zeta_2, \varphi_1, \varphi_2$,

$$\begin{aligned}(h_1, \zeta_1, \varphi_1) \prec_n (h_2, \zeta_2, \varphi_2) \quad &\text{if} \quad h_1 < h_2 \leq h_n, \quad \text{or if} \quad h_n < h_1 < h_2, \\ &\text{or if} \quad h_2 \leq h_n < h_1.\end{aligned} \tag{4.18}$$

By the definition of the Coxian distributions and the service discipline, it follows that phase transitions without arrivals or departures of jobs are only possible to states with a higher order with respect to these orderings, and that from each reachable state (n, h, ζ, φ), $n \neq 0$, there exists a path of states of increasing order with respect to $\prec_n$ which leads with positive probability to a state $(n, h_0, 1, 1)$ for some $h_0 \in \{1, \ldots, L\}$ from which a departure is possible. Note that the state of highest order is $(n, h_n, 1, 1)$ and that this state is reachable by definition of h_n.

Based on the foregoing considerations, the following power-series expansions can be introduced, cf. (2.5), for $n \in \mathbb{N}^s$, $h = 1, \ldots, L$, $\zeta = 0, 1$, $\varphi = 1, \ldots, \Psi_h^0$ if $\zeta = 0$, $\varphi = 1, \ldots, \Psi_{l(h)}^1$ if $\zeta = 1$,

$$p(n, h, \zeta, \varphi) = \chi^{n_1 + \cdots + n_s} \sum_{k=0}^{\infty} \chi^k b(k; n, h, \zeta, \varphi). \tag{4.19}$$

As indicated in section 2.4, the following set of equations follows from the balance equations (4.13), (4.14): for $n \in \mathbb{N}^s$, $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$, $k = 0, 1, \ldots,$

$$\mu_h^{0,\varphi} b(k;n,h,0,\varphi) = \mu_h^{0,\varphi+1} b(k;n,h,0,\varphi+1) I\{\varphi < \Psi_h^0\}$$

$$+ \sum_{j=1}^{s} a_j \big[b(k;n-e_j,h,0,\varphi) I\{n_j > 0\} - b(k-1;n,h,0,\varphi) I\{k > 0\} \big]$$

$$+ \mu_{l(h-1)}^{1,1} \pi_h^{0,\varphi} b(k-1;n+e_{l(h-1)},h-1,1,1) I\{k > 0\} \big[1 - q_{h-1} I\{n_{l(h-1)} > 0\} \big]$$

$$+ \mu_{h-1}^{0,1} \pi_h^{0,\varphi} b(k;n,h-1,0,1) I\{n_{l(h-1)} = 0\}; \tag{4.20}$$

for $n \in \mathbb{N}^s$, $h = 1, \ldots, L$, $j = 1, \ldots, \Psi_{l(h)}^1$, $n_{l(h)} > 0$, $k = 0, 1, \ldots$,

$$\mu_{l(h)}^{1,\varphi} b(k;n,h,1,\varphi) = \mu_{l(h)}^{1,\varphi+1} b(k;n,h,1,\varphi+1) I\{\varphi < \Psi_{l(h)}^1\}$$

$$+ \mu_h^{0,1} \pi_{l(h)}^{1,\varphi} b(k;n,h,0,1) + q_h \mu_{l(h)}^{1,1} \pi_{l(h)}^{1,\varphi} b(k-1;n+e_{l(h)},h,1,1) I\{k > 0\}$$

$$+ \sum_{j=1}^{s} a_j \big[b(k;n-e_j,h,1,\varphi) I\{n_j > 0\} - b(k-1;n,h,1,\varphi) I\{k > 0\} \big]. \tag{4.21}$$

This set of eqs. (4.20), (4.21) forms a recursive scheme for all coefficients $b(k; n, h, \zeta, \varphi)$ except those of states with $n = 0$, because eqs. (4.20), (4.21) express the coefficients $b(k; n, h, \zeta, \varphi)$ in terms of coefficients with a lower order, either with respect to the partial ordering $\prec$ defined in (2.7) or with respect to the orderings defined in (4.16) and (4.18). Hence, the only states which require further attention are states with $n = 0$ and $\zeta = 0$. For these states, eqs. (4.20) read: for $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$, $k = 0, 1, \ldots$,

$$\mu_h^{0,\varphi} b(k;0,h,0,\varphi) = \mu_h^{0,\varphi+1} b(k;0,h,0,\varphi+1) I\{\varphi < \Psi_h^0\}$$

$$+ \mu_{h-1}^{0,1} \pi_h^{0,\varphi} b(k;0,h-1,0,1) + y(k;h,\varphi); \tag{4.22}$$

here, the quantities $y(k; h, \varphi)$, $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$, defined by $y(0; h, \varphi) := 0$ and for $k = 1, 2, \ldots$ by

$$y(k;h,\varphi) := \mu_{l(h-1)}^{1,1} \pi_h^{0,\varphi} b(k-1;e_{l(h-1)},h-1,1,1)$$

$$- \sum_{j=1}^{s} a_j b(k-1;0,h,0,\varphi), \tag{4.23}$$

consist of terms with coefficients of lower order with respect to $\prec$ than $(k; 0, h, 0, \varphi)$, cf. (2.7), and, hence, can be considered to be known. The sets of eqs. (4.22) are, for k fixed, dependent, as in the general case, cf. (2.8). The law of total probability gives (cf. (2.9)),

$$\sum_{h=1}^{L}\sum_{\varphi=1}^{\Psi_h^0} b(k;0,h,0,\varphi) = \begin{cases} 1, & \text{for } k=0, \\ -Y(k), & \text{for } k=1,2,\ldots. \end{cases} \tag{4.24}$$

Here, for $k = 1, 2, \ldots,$

$$Y(k) := \sum_{0<n_1+\ldots+n_s\le k}\cdots\sum \sum_{h=1}^{L}\sum_{\zeta=0}^{1}\sum_{\varphi=1}^{\Psi_h^\zeta} b(k-n_1-\ldots-n_s;n,h,\zeta,\varphi). \tag{4.25}$$

Consider, for k fixed, the set of equations consisting of (4.24) and all but one of the equations (4.22). It is readily verified that the determinants $\Delta(k)$ of these sets of equations are independent of k and are given by:

$$\Delta(k) = \Delta := \sigma_1 \prod_{h=1}^{L}\prod_{\varphi=1}^{\Psi_h^0} \mu_h^{0,\varphi}, \quad \text{for } k=0,1,2,\ldots. \tag{4.26}$$

For $k = 0$, this set of equations is readily solved: for $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$,

$$b(0;0,h,0,\varphi) = \frac{1}{\sigma_1 \mu_h^{0,\varphi}} \sum_{\psi=\varphi}^{\Psi_h^0} \pi_h^{0,\psi}. \tag{4.27}$$

It is more tedious, but straightforward, to show that for $k = 1, 2, \ldots,$

$$b(k;0,L,0,1) = \frac{-1}{\sigma_1 \mu_L^{0,1}}\left[Y(k) + \sum_{h=1}^{L}\sum_{\varphi=1}^{\Psi_h^0}\frac{1}{\mu_h^{0,\varphi}}\sum_{\psi=\varphi}^{\Psi_h^0}\left\{y(k;h,\psi) + \pi_h^{0,\psi}\sum_{j=1}^{h-1}\sum_{\nu=1}^{\Psi_j^0} y(k;j,\nu)\right\}\right]. \tag{4.28}$$

Once the coefficient $b(k; 0, L, 0, 1)$ has been determined according to (4.28), the other coefficients $b(k; 0, h, 0, \varphi)$, $h = 1, \ldots, L$, $\varphi = 1, \ldots, \Psi_h^0$, can be sequentially obtained with the aid of (4.22). Hence, relations (4.27), (4.20), (4.21), (4.28) and (4.22) form a complete scheme for computing the coefficients of the power-series expansions of the state probabilities.

The number of multiplications and divisions required to compute coefficients $b(k; n, h, \zeta, \varphi)$ for all ζ and φ for $k + n_1 + \ldots + n_s \le M$ for some M is (in the case $G = 0$) roughly given by (cf. (4.20), (4.21)),

$$\binom{M+s+1}{s+1}\left[s(s-1) + (s+4)\sum_{h=1}^{L}\Psi_h^0 + (s+7)\sum_{h=1}^{L}\Psi_{l(h)}^1 - 2L\right].$$

Here, the fact that terms vanish in (4.20) and (4.21) when $k, n_1, \ldots, n_s$ are not all positive has been ignored. It should be noted that this number of operations can be reduced by storing products like $q_h \mu_{l(h)}^{1,1} \pi_{l(h)}^{1,\varphi}$, cf. (4.21), in separate arrays. This number of operations is considerably less than the upper bound (2.23) for $G = 0$, where $|\Theta|$ is given by (3.1) with $K_h = 1$ for $h = 1, \ldots, L$ and with $\Psi_{l(h-1),l(h)}^{0}$ denoting Ψ_h^0.

4.5. WAITING TIMES

Let W_j denote a random variable distributed as the stationary waiting time of jobs arriving at queue j, $j = 1, \ldots, s$. The number of jobs at queue j left behind by a job departing from the queue is equal to the number of jobs that arrived at queue j during the sojourn time of the departing job. Because arrivals occur according to a Poisson process, this implies (cf. Takagi [25]),

$$E\{z^{N_j}\} = E\{e^{-\lambda_j(1-z)W_j}\}\ \gamma_j(\lambda_j(1-z)), \quad |z| \le 1, j = 1, \ldots, s, \tag{4.29}$$

cf. (4.1). The moments of the waiting time distributions can be obtained from the moments of the marginal queue length distributions through these relations. The expected values of the waiting times for jobs in the various queues of a system with a cyclic polling order satisfy the following pseudo-conservation law (cf. Tedijanto [27]),

$$\sum_{j=1}^{s}\left[1 - a_j(1-q_j)\frac{\sigma_1 \chi}{1-\rho}\right]\eta_j E\{W_j\}$$

$$= \frac{\rho}{1-\rho}\frac{\beta_2}{2\beta_1} + \frac{\sigma_2}{2\sigma_1} + \frac{\sigma_1 \rho}{1-\rho}\left[\sum_{j=1}^{s}\eta_j^2(1-q_j) + \frac{1}{2}\sum_{j=1}^{s}\eta_j(1-\eta_j)\right]. \tag{4.30}$$

Here, $\eta_j := \rho_j/\rho$ is the relative offered load at queue j, $j = 1, \ldots, s$. This relation provides a useful check on the accuracy of the computations. For systems with non-cyclic periodic order-of-visit rules, a pseudo-conservation law is only known for the special case of exhaustive, gated and 1-limited number-of-services rules (i.e. $q_h = 1$ or $q_h = 0$, $h = 1, \ldots, L$, in the present model), with the restriction that queues with 1-limited service may be listed only once on the polling table, cf. Boxma et al. [10].

An important general property of the waiting times in systems with fixed polling orders is the following heavy traffic behaviour: for $j = 1, \ldots, s$, $E\{W_j\}$ tends to infinity as $x \uparrow 1$ if and only if (cf. (4.6), (4.10), (4.11)),

$$a_j/m_j = \max_{i=1,\ldots,s}\{a_i/m_i\}. \tag{4.31}$$

This implies that the modifications (2.18) and (2.19) for the initial sequence of the epsilon algorithm should only be applied to moments of distributions related to queues for which (4.31) holds. That only the arrival rates, and not the service rates, play a role in condition (4.31) can be explained by the fact that a certain (integer) number of jobs is served during each cycle of the server along the queues according to the polling table and the Bernoulli schedules. We will call service disciplines for which the mean waiting time at each station tends to infinity as the occupancy tends to one *balanced* disciplines. When the polling order for a system is fixed, then, in order that a discipline is balanced, the mean number of services attributed to the queues during one period must be such that

$$m_j : m_i = a_j : a_i, \qquad \text{for } i, j = 1, \ldots, s. \tag{4.32}$$

4.6. EXAMPLE: INFLUENCE OF THE MOMENTS OF THE SERVICE TIME DISTRIBUTIONS

The aim of the examples in this section is to study the influence of higher-order moments of the service time distributions on the mean and the standard deviation of the waiting times. For this purpose, we consider systems with four stations, with equal arrival rates, equal mean service times, cyclic polling, and equal switching rates between the stations, for various service time distributions. Four distributions will be considered, labeled A, B, C and D, each with mean $\gamma_{1j} = 1$, $j = A, B, C, D$. Distributions A and C have second moments $\gamma_{2j} = 3$, $j = A, C$, and consist both of two phases; these distributions differ in the third moment: $\gamma_{3A} = 15$, $\gamma_{3C} = 20.25$. Distributions B and D have second moment $\gamma_{2j} = 1.75$, $j = B, D$; distribution B consists of four phases, $\gamma_{3B} = 3.98$, while distribution D consists of three phases, $\gamma_{3D} = 6$. The arrival rates are: $\lambda_j = 0.2$, $j = 1, 2, 3, 4$, and the switching times are identically, exponentially distributed, with $\sigma_1 = 0.1$. Table 4 shows the mean and the standard deviation of the waiting times in several of such 4-station systems. The stations have either all 1-limited service (L) or all exhaustive service (E), and the stations are visited in cyclic order. Under the heading "system" it is indicated that the system consists either of four stations with different service time distributions ("ABCD", visited in this order), or consists of two pairs of stations with different distributions ("ACAC/BDBD" indicates that the data for W_A and W_C stem from a system with two stations with distribution A and two stations with distribution C, visited in the order ACAC; and analogously for W_B and W_D), or consists of four stations with identical distributions ("symmetrical" indicates that the data for W_A stem from a system with four stations with distribution A; and analogously for W_B, W_C and W_D). The numerical results indicate that the vth moments of the service time distributions have an impact on the overall level of the $(v-1)$th moments of the waiting time distributions, as in the case of the *M/G/1* queue, but that the influence of the moments of the service time distribution at a certain queue on the moments of the waiting time distribution at that queue is less important. It has been observed that the relative differences between W_A and W_C, respectively W_B and W_D,

Table 1

The influence of higher-order moments of the service time distributions on the waiting time distributions

System	DCP	$E\{W_A\}$	$E\{W_B\}$	$E\{W_C\}$	$E\{W_D\}$	$\sigma\{W_A\}$	$\sigma\{W_B\}$	$\sigma\{W_C\}$	$\sigma\{W_D\}$
ABCD	L	5.657	5.598	5.644	5.601	8.220	8.085	8.231	8.082
ABAB/CDCD	L	5.651	5.599	5.650	5.600	8.039	7.894	8.408	8.268
ACAC/BDBD	L	7.018	4.239	7.010	4.234	10.081	5.997	10.086	6.003
symmetrical	L	7.014	4.236	7.014	4.236	9.867	5.855	10.295	6.142
ABCD	E	4.873	5.052	4.873	5.052	7.285	7.817	7.272	7.673
ABAB/CDCD	E	4.873	5.052	4.873	5.052	7.076	7.523	7.476	7.962
ACAC/BDBD	E	6.213	3.713	6.213	3.713	9.459	5.413	9.374	5.356
symmetrical	E	6.213	3.713	6.213	3.713	9.172	5.219	9.655	5.545

decrease when σ_1 increases. In the case of exhaustive service, the mean waiting times depend only on the first two moments of the service time distributions, as can be seen from the set of equations (given in Baker and Rubin [1] for general polling tables) which determine these quantities.

4.7. EXAMPLE: THE INFLUENCE OF THE SERVICE DISCIPLINES

The aim of this section is to study the influence of the Bernoulli parameters and of the polling order on the mean waiting times in a given asymmetrical system consisting of four stations, and to find the optimal values of the Bernoulli parameters for this system with respect to some cost functions which are linear functions of the mean waiting times. The parameters of the stations are listed in table 5. The

Table 5

Parameters and cost coefficients for a system with four stations

j	λ_j	γ_{1j}	γ_{2j}	γ_{3j}	ρ_j	c_{1j}	c_{2j}	c_{3j}	c_{4j}	c_{5j}
1	0.128	1.00	2.00	6.00	0.128	0.25	0.10	0.16	0.10	0.04
2	0.128	0.25	0.31	0.70	0.032	0.25	0.10	0.04	0.40	0.16
3	0.512	1.00	1.75	4.28	0.512	0.25	0.40	0.64	0.10	0.16
4	0.512	0.25	0.09	0.05	0.128	0.25	0.40	0.16	0.40	0.64

service time distributions are exponential (station 1) or consist of two exponential phases, so that they are completely determined by their first three moments. The five cost functions that we consider for this system are:

$$C_i := \sum_{j=1}^{s} c_{ij} E\{W_j\}, \quad i = 1, \ldots, 5. \tag{4.33}$$

Here, the coefficients of the cost functions are defined by:

$$c_{1j} := \frac{1}{s}, \quad c_{2j} := \frac{\lambda_j}{\Lambda}, \quad c_{3j} := \eta_j = \frac{\rho_j}{\rho},$$

$$c_{4j} := \frac{1}{\gamma_{1j}}\left[\sum_{k=1}^{s}\frac{1}{\gamma_{1k}}\right]^{-1}, \quad c_{5j} := \frac{\lambda_j}{\gamma_{1j}}\left[\sum_{k=1}^{s}\frac{\lambda_k}{\gamma_{1k}}\right]^{-1}, \quad j = 1, \ldots, s. \tag{4.34}$$

Cost functions C_1, C_2 and C_3 are, respectively, station-, job- and load-weighted averages of the mean waiting times; cost functions C_4, C_5 and C_2 are, respectively, station-, job- and load-weighted averages of the mean waiting times divided by the corresponding mean service times.

First, we consider cyclic polling strategies: the stations are visited in the order 1, 2, 3, 4. Usually, the order in which stations are arranged in a cycle has only a minor influence on the waiting times, cf. Blanc [4]. The switching times are identically, Erlang-2 distributed, with $\delta_{1j} = 0.05$, $j = 1, 2, 3, 4$, so that $\sigma_1 = 0.2$. Table 6 contains a list of Bernoulli schedules for this cyclic polling strategy, and the corresponding values of the mean waiting times and the cost functions can be found in table 7. The schedules Da*, Db*, Dc*, Dd1 and Dd4 are extremal in the space of Bernoulli schedules (q_1, q_2, q_3, q_4), $0 \le q_j \le 1$, $j = 1, 2, 3, 4$. Intuitively, $E\{W_v\}$ is minimal for Dav and maximal for Dbv over all Bernoulli schedules for cyclic polling orders. The Bernoulli probabilities are the same for each queue in the schedules Dd*; the schedules De* and Dd4 are balanced disciplines, cf. (4.32). The schedules Df* are such that the maximal mean visit time to queue j, $m_j \gamma_{1j}$, is the same for each queue ($j = 1, 2, 3, 4$). For the schedules Dg*, the maximal mean number of services is proportional to the load of a queue, i.e. $m_i : m_j = \rho_i : \rho_j$, $i, j = 1, 2, 3, 4$. The schedules Dh* have been determined by an extensive search. It is conjectured that cost function C_v is minimal over all Bernoulli schedules for the considered cyclic polling order for Dhv, $v = 1, \ldots, 5$, where Dh3 = Dd4. That the purely exhaustive discipline Dd4 is optimal for cost function C_3 is obvious from the pseudo-conservation law (4.30). It is interesting to note that in Blanc [7], light traffic asymptotes of the mean waiting times have been determined by algebraic evaluation of the computation scheme of the PSA for the special case of cyclic polling, exponential service times and negligible switching times. These asymptotes indicate that it is optimal in light traffic to take for cost function C_i, $i = 1, 2, 4, 5$,

$$q_j = 1, \text{ if } \eta_j < c_{ij}, \quad q_j = 0, \text{ if } \eta_j > c_{ij}, \qquad j = 1, \ldots, 4. \tag{4.35}$$

The numerical results for the present example with non-exponential service times and non-negligible switching times suggest that it is still optimal to take $q_j = 1$ for queues with $\eta_j < c_{ij}$, but that q_j should increase with increasing load for queues with $\eta_j > c_{ij}$, $j = 1, 2, 3, 4$. This is supported by the stability condition (4.10), which

Bernoulli schedules for the model of table 7

DCP	q_1	q_2	q_3	q_4	χ/ρ	DCP	q_1	q_2	q_3	q_4	χ/ρ
Da1	1.00	0.00	0.00	0.00	1.1280	Db1	0.00	1.00	1.00	1.00	1.0320
Da2	0.00	1.00	0.00	0.00	1.1280	Db2	1.00	0.00	1.00	1.00	1.0320
Da3	0.00	0.00	1.00	0.00	1.1280	Db3	1.00	1.00	0.00	1.00	1.1280
Da4	0.00	0.00	0.00	1.00	1.1280	Db4	1.00	1.00	1.00	0.00	1.1280
Dc1	1.00	1.00	0.00	0.00	1.1280	Dc4	0.00	0.00	1.00	1.00	1.0320
Dc2	1.00	0.00	1.00	0.00	1.1280	Dc5	0.00	1.00	0.00	1.00	1.1280
Dc3	1.00	0.00	0.00	1.00	1.1280	Dc6	0.00	1.00	1.00	0.00	1.1280
Dd1	0.00	0.00	0.00	0.00	1.1280	Dd3	0.95	0.95	0.95	0.95	1.0064
Dd2	0.80	0.80	0.80	0.80	1.0256	Dd4	1.00	1.00	1.00	1.00	1.0000
De1	0.20	0.20	0.80	0.80	1.0256	De2	0.80	0.80	0.95	0.95	1.0064
Df1	0.20	0.80	0.20	0.80	1.1024	Df2	0.80	0.95	0.80	0.95	1.0256
Dg1	0.80	0.20	0.95	0.80	1.0256	Dg2	0.95	0.80	0.99	0.95	1.0064
Dh1	1.00	1.00	0.50	1.00	1.0640	Dh4	0.45	1.00	0.20	1.00	1.1024
Dh2	0.15	1.00	0.75	1.00	1.0320	Dh5	0.00	1.00	0.35	1.00	1.0832

Table 7

System with four stations with an offered load of $\rho = 0.80$ and with equal switching rates between the stations: cyclic polling order

DCP	$E\{W_1\}$	$E\{W_2\}$	$E\{W_3\}$	$E\{W_4\}$	C_1	C_2	C_3	C_4	C_5
Da1	1.06	1.59	9.52	7.43	4.90	7.04	7.51	4.66	6.57
Da2	1.63	1.12	9.44	7.36	4.89	7.00	7.53	4.50	6.47
Da3	5.52	5.03	1.54	19.68	7.94	9.54	5.22	10.59	13.87
Da4	1.84	1.69	10.53	1.08	3.78	5.00	7.27	2.34	2.72
Db1	10.52	4.20	1.89	3.77	5.10	3.74	3.66	4.43	3.81
Db2	4.70	13.05	2.33	4.57	6.16	4.53	3.50	7.75	5.57
Db3	1.17	1.32	10.66	1.16	3.58	4.98	7.25	2.17	2.70
Db4	3.37	3.83	1.82	20.41	7.36	9.61	5.12	10.22	14.10
Dc1	1.06	1.20	9.53	7.44	4.81	7.02	7.51	4.51	6.52
Dc2	3.29	6.25	1.76	20.26	7.89	9.76	5.14	11.11	14.38
Dc3	1.17	1.80	10.64	1.15	3.69	5.01	7.25	2.36	2.77
Dc4	9.89	9.02	1.82	3.67	6.10	4.09	3.70	6.25	4.48
Dc5	1.86	1.23	10.55	1.08	3.68	4.96	7.27	2.16	2.65
Dc6	5.71	3.16	1.56	19.78	7.55	9.42	5.21	9.90	13.64
Dd1	1.62	1.50	9.43	7.35	4.98	7.02	7.53	4.65	6.52
Dd2	3.11	3.32	3.75	5.11	3.82	4.19	3.85	4.06	4.53
Dd3	4.28	4.77	2.87	5.06	4.24	4.07	3.52	4.64	4.63
Dd4	5.01	5.65	2.52	4.90	4.52	4.04	3.43	4.98	4.65
De1	5.39	5.03	3.33	4.51	4.57	4.18	3.92	4.69	4.44
De2	5.10	5.46	2.72	4.78	4.51	4.05	3.54	4.88	4.57
Df1	1.97	1.49	8.28	1.67	3.35	4.33	5.94	2.29	2.71
Df2	3.37	3.20	4.10	3.15	3.46	3.56	3.79	3.29	3.32
Dg1	4.20	8.11	2.35	7.26	5.48	5.08	3.66	6.80	6.49
Dg2	4.71	6.53	2.44	5.65	4.83	4.36	3.48	5.58	5.24
Dh1	1.80	2.03	6.18	1.74	2.94	3.55	4.60	2.31	2.50
Dh2	5.48	2.64	4.06	2.33	3.63	3.37	3.95	2.94	2.79
Dh4	1.79	1.45	8.40	1.27	3.23	4.19	5.92	2.11	2.46
Dh5	2.69	1.60	7.02	1.41	3.18	3.80	5.21	2.17	2.39

implies that the purely exhaustive discipline Dd4 will outperform any Bernoulli discipline for any system with cyclic polling when the offered load is sufficiently high, provided that all cost coefficients are positive.

Next, we consider more general periodic visit orders for the same system. Table 8 contains a list of polling tables and Bernoulli schedules. The schedules D*B are balanced disciplines, cf. section 4.5. The Bernoulli parameters of stations which appear more than once on the polling table have been taken to be the same for each visit. For the discipline DkB, we have also considered two variants: in discipline DkBP, the Bernoulli parameters of stations 3 and 4 are such that the proportion between the average maximal duration of an intervisit time and that of the subsequent visit is the same for each visit; in DkBI, they are such that the average maximal durations of the intervisit times are the same for each station. The schedules D*L and D*E in table 9 are 1-limited ($q_h = 0, h = 1, \ldots, L$), respectively exhaustive ($q_h = 1, h = 1, \ldots, L$) disciplines with the same polling table as the corresponding discipline D*B. For all these disciplines, the switching times between any pair of stations are identically, Erlang-2 distributed, with $\delta_{1j} = 0.05$, $j = 1, \ldots, L$, so that $\sigma_1 = 0.05 \times L$, except for the scan-type disciplines Dm* and Do*, where the switching times between two consecutive visits to the same station have been taken negligibly small so that $\sigma_1 = 0.3$ in these cases. The "end"-stations in the cases DmL and DoL have in fact a 2-limited service rule; for comparison, we have inserted disciplines DnD and DpD, respectively, in which the "end"-stations have Bernoulli parameters 0.5 and the intermediate stations have 1-limited service. For the disciplines Dk* and Dl*, also a variant is considered in which it is assumed that the stations are arranged in a cycle and that the mean switching times between stations 1 and 3, respectively 2 and 4, are twice as long as those between the other, adjacent, pairs of stations; hence, $\sigma_1 = 0.4$ for the disciplines Dk*C and $\sigma_1 = 0.5$ for the disciplines Dl*C.

Table 8

Polling tables and Bernoulli schedules for the model of table 9

DCP	Table	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	χ/ρ
DiB	132343	0.60	0.70	0.60	0.70	0.90	0.70			1.0192
DjB	13432343	0.60	0.60	0.80	0.60	0.60	0.60	0.80	0.60	1.0256
DkBP	134234	0.60	0.83	0.84	0.60	0.75	0.72			1.0192
DkBI	134234	0.60	0.75	0.89	0.60	0.83	0.20			1.0192
DkB	134234	0.60	0.80	0.80	0.60	0.80	0.80			1.0192
DlB	13432434	0.60	0.70	0.70	0.70	0.60	0.70	0.70	0.70	1.0256
DmB	31244213	0.80	0.20	0.20	0.80	0.80	0.20	0.20	0.80	1.0192
DnB	312421	0.90	0.20	0.20	0.90	0.20	0.20			1.0192
DoB	13422431	0.20	0.80	0.80	0.20	0.20	0.80	0.80	0.20	1.0192
DpB	134243	0.60	0.80	0.80	0.60	0.80	0.80			1.0192

Table 9

System with four stations with an offered load of $\rho = 0.80$ and with equal switching rates between the stations: non-cyclic polling orders

DCP	$E\{W_1\}$	$E\{W_2\}$	$E\{W_3\}$	$E\{W_4\}$	C_1	C_2	C_3	C_4	C_5
DiL	3.61	3.22	3.16	42.23	13.05	18.84	9.49	18.86	28.19
DiB	5.87	5.85	2.76	5.32	4.95	4.40	3.79	5.33	5.01
DiE	6.04	6.53	1.93	6.07	5.14	4.46	3.43	5.84	5.48
DjL	5.91	5.27	3.30	14.17	7.16	8.11	5.54	8.70	10.68
DjB	6.63	6.69	2.99	4.30	5.15	4.25	3.93	5.36	4.57
DjE	6.91	7.94	1.92	4.96	5.43	4.24	3.45	6.04	5.03
DkL	3.17	2.86	6.81	5.32	4.54	5.45	5.83	4.27	5.08
DkLC	3.53	3.20	9.11	7.11	5.74	7.16	7.66	5.39	6.66
DkBP	6.02	6.13	2.98	4.11	4.81	4.05	3.78	5.00	4.33
DkBI	5.91	6.00	2.93	4.56	4.85	4.19	3.79	5.11	4.59
DkB	6.06	6.13	2.98	4.05	4.81	4.03	3.77	4.98	4.29
DkBC	6.52	6.59	3.17	4.30	5.14	4.30	4.02	5.32	4.58
DkE	6.39	7.23	2.19	4.41	5.06	4.00	3.42	5.52	4.59
DkEC	6.62	7.48	2.26	4.54	5.22	4.13	3.53	5.70	4.73
DlL	5.59	4.96	5.80	4.53	5.22	5.19	5.53	4.93	4.84
DlLC	6.33	5.63	6.97	5.44	6.09	6.16	6.57	5.75	5.75
DlB	6.71	6.71	3.12	3.76	5.08	4.09	3.94	5.17	4.25
DlBC	7.19	7.21	3.29	3.97	5.42	4.35	4.18	5.52	4.51
DlE	7.00	7.82	2.09	4.30	5.30	4.04	3.46	5.76	4.62
DlEC	7.23	8.10	2.14	4.39	5.46	4.15	3.55	5.93	4.74
DmL	1.60	1.62	7.09	5.81	4.03	5.48	5.79	3.84	5.17
DmB	4.90	4.81	3.19	4.81	4.43	4.17	3.79	4.66	4.55
DnL	1.07	1.11	21.81	16.98	10.24	15.73	16.89	9.52	14.57
DnD	1.82	1.83	7.03	7.12	4.45	6.02	6.00	4.47	6.05
DnB	4.63	4.57	3.30	5.31	4.45	4.36	3.89	4.75	4.84
DnE	3.86	4.78	2.84	5.66	4.28	4.26	3.53	4.85	4.99
DoL	1.84	1.87	7.04	5.69	4.11	5.46	5.79	3.91	5.14
DoB	5.28	5.30	3.04	4.68	4.58	4.15	3.75	4.82	4.54
DpL	3.15	2.86	6.80	5.52	4.58	5.53	5.85	4.35	5.20
DpD	2.17	2.17	6.98	5.64	4.24	5.48	5.80	4.04	5.16
DpB	5.91	6.05	2.94	4.50	4.85	4.17	3.79	5.11	4.56
DpE	5.82	6.81	2.11	5.63	5.09	4.36	3.45	5.77	5.26

5. Conclusions

A general framework for application of the PSA has been described in section 2. The PSA has been discussed in detail for a broad class of polling systems with periodic visit orders and Bernoulli schedules in section 4. Many polling systems with other service disciplines also fit in the general setting introduced in section 2. Examples are, cf. section 3, limited service (B3), cf. Blanc [7], and Markovian (A2) and dynamic (A4) order-of-visit rules. Another possible extension consists of Coxian interarrival times, but they have a strong impact on the size of the supplementary

space Θ: the sizes indicated in (3.1)–(3.4) should by *multiplied* by the factors Ψ_j^a, the number of phases of the interarrival times at station j, for $j = 1, \ldots, s$. Other variants are models with finite buffers which require a few minor modifications, e.g. concerning the definition of the occupancy χ, if all buffers are finite and the system is stable for any offered load, cf. Blanc [5]. Less straightforward generalizations of the algorithm are needed for models with batch arrivals: then the birth–death structure is violated and the key property (2.3) does not hold. It seems to be possible to obtain a recursive set of equations by introducing power-series expansions as functions of some root of χ in the case of bounded batch sizes, but the applicability will be limited to small batch sizes because the required number of coefficients will grow very rapidly with the batch sizes. A better alternative might be to consider models with Markov modulated arrival processes, although they have a similar impact on the size of the supplementary space as Coxian interarrival times, while they disturb the recursive character of the computation scheme. A final extension of the PSA which we mention here is the addition of migration to the processes, i.e. the admission of transitions to states $(n + e_j - e_i, \psi)$, $i, j = 1, \ldots, s$, $\psi \in \Theta$, $n_i > 0$, from a state $(n, \varphi) \in \mathbb{N}^s \times \Theta$, which makes it possible to model networks of queues in which jobs may move from one queue to another queue. The computation scheme will only be recursive if the network is acyclic, as will be shown in a future paper.

References

[1] J.E. Baker and I. Rubin, Polling with a general-service order table, IEEE Trans. Commun. COM-35(1987)283–288.

[2] J.P.C. Blanc, A note on waiting times in systems with queues in parallel, J. Appl. Probab. 24(1987) 540–546.

[3] J.P.C. Blanc, On a numerical method for calculating state probabilities for queueing systems with more than one waiting line, J. Comput. Appl. Math. 20(1987)119–125.

[4] J.P.C. Blanc, A numerical study of a coupled processor model, in: *Computer Performance and Reliablity*, ed. G. Iazeolla, P.J. Courtois and O.J. Boxma (North-Holland, Amsterdam, 1988), pp. 289–303.

[5] J.P.C. Blanc, A numerical approach to cyclic-service queueing models, Queueing Systems 6(1990) 173–188.

[6] J.P.C. Blanc, The power-series algorithm applied to the shortest-queue model, Report FEW 379, Department of Economics, Tilburg University (1989), Oper. Res. (1992), to appear.

[7] J.P.C. Blanc, Cyclic polling systems: Limited service versus Bernoulli schedules, Report FEW 422, Department of Economics, Tilburg University (1990).

[8] J.P.C. Blanc, The power-series algorithm applied to cyclic polling systems, Report FEW 445, Department of Economics, Tilburg University (1990), Stoch. Models 7(1991), to appear.

[9] O.J. Boxma, Workloads and waiting times in single-server systems with multiple customer classes, Queueing Systems 5(1989)185–214.

[10] O.J. Boxma, W.P. Groenendijk and J.A. Weststrate, A pseudo-conservation law for service systems with a polling table, IEEE Trans. Commun. COM-38(1990)1865–1870.

[11] O.J. Boxma and J.A. Weststrate, Waiting times in polling systems with Markovian server routing, in: *Messung, Modellierung und Bewertung von Rechensystemen und Netzen*, ed. G. Stiege and J.S. Lie (Springer, Berlin, 1989), pp. 89–104.

[12] C. Brezinski, *Accélération de la Convergence en Analyse Numérique*, Lecture Notes in Mathematics 584 (Springer, Heidelberg, 1977).
[13] J.W. Cohen, A two-queue model with semi-exhaustive alternating service, in: *Performance '87*, ed. P.-J. Courtois and G. Latouche (North-Holland, Amsterdam, 1988), pp. 19–37.
[14] J.W. Cohen, A two-queue, one-server model with priority for the longer queue, Queueing Systems 2(1987)261–283.
[15] R.B. Cooper, Queues served in cyclic order: Waiting times, Bell. Syst. Tech. J. 49(1970)399–413.
[16] M. Eisenberg, Queues with periodic service and changeover times, Oper. Res. 20(1972)440–451.
[17] G. Hooghiemstra, M. Keane and S. van de Ree, Power series for stationary distributions of coupled processors models, SIAM J. Appl. Math. 48(1988)1159–1166.
[18] N.K. Jaiswal, *Priority Queues* (Academic Press, New York, 1968).
[19] L. Kleinrock and H. Levy, The analysis of random polling systems, Oper. Res. 36(1988)716–732.
[20] G.P. Klimov and G.K. Mishkoy, *Priority Service Systems with Orientation* (Moscow University Press, Moscow, 1979), in Russian.
[21] P.J. Kühn, Multi-queue systems with non-exhaustive cyclic service, Bell Syst. Tech. J. 58(1979) 671–698.
[22] H. Levy and M. Sidi, Polling systems: Applications, modeling, and optimization, IEEE Trans. Commun. COM-38(1990)1750–1760.
[23] D. Sarkar and W.I. Zangwill, Expected waiting time for non-symmetric cyclic queueing systems – exact results and applications, Manag. Sci. 35(1989)1463–1474.
[24] L.D. Servi, Average delay approximation of *M/G/*1 cyclic service queues with Bernoulli schedules, IEEE J. Sel. Areas Comm. SAC-4(1986)813–822.
[25] H. Takagi, *Analysis of Polling Systems* (The MIT Press, Cambridge, MA, 1986).
[26] H. Takagi, Queueing analysis of polling models: An update, in: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, Amsterdam, 1990), pp. 267–318.
[27] Tedijanto, Exact results for the cyclic-service queue with a Bernoulli schedule, Perf. Eval. 11(1990) 107–115.
[28] P. Wynn, On the convergence and stability of the epsilon algorithm, SIAM J. Numer. Anal. 3(1966) 91–122.